



KOREAN PATENT ABSTRACTS(KR)

Document Code:A

(11) Publication No.1020010084661

(43) Publication.Date. 20010906

(21) Application No.1020000009856

(22) Application Date. 20000228

(51) IPC Code:

H04N 7/28

(71) Applicant:

LG ELECTRONICS INC.

(72) Inventor:

HONG, SEONG HUN

(30) Priority:

(54) Title of Invention

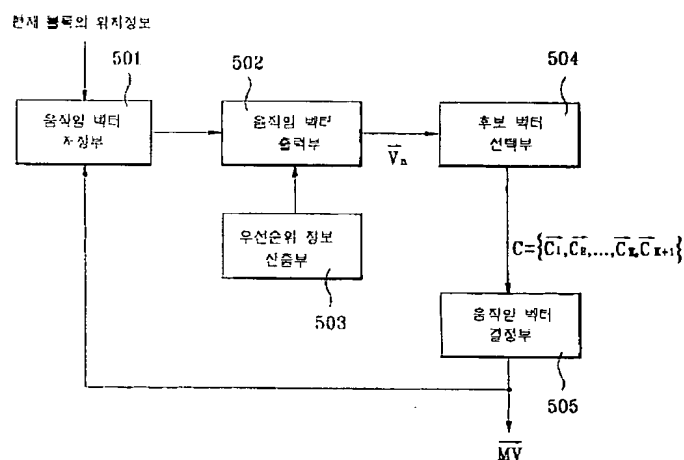
MOTION ESTIMATING METHOD AND APPARATUS

Representative drawing

(57) Abstract:

PURPOSE: A motion estimating method and apparatus are provided to obtain a fast convergence time and an exact motion vector.

CONSTITUTION: A motion vector storage section(501) stores a motion vector with respect to each block of a previous frame and a motion vector of a block a motion vector of which is previously calculated among blocks of a current frame. A motion vector output section(502) outputs from a motion vector having a high priority order in motion vectors output from the motion vector storage section (501). A priority order information calculator(503) determines a priority order of the motion vectors. A candidate vector selector(504) selects (K+1) different candidate



vectors from the motion vectors from the motion vector output section(502). A

motion vector determining section(505) determines a vector which minimizes an estimate function among the (K+1) different candidate vectors as a motion vector of a current block.

COPYRIGHT 2001 KIPO

if display of image is failed, press (F5)

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) Int. Cl. ⁷	(11) 공개번호	특2001-0084661
H04N 7/28	(43) 공개일자	2001년09월06일

(21) 출원번호	10-2000-0009856
(22) 출원일자	2000년02월28일
(71) 출원인	엘지전자주식회사
	서울시영등포구여의도동20번지
(72) 발명자	홍성훈
	서울특별시서초구양재동우성아파트109동206호
(74) 대리인	김용인, 심창섭

심사청구 : 있음

(54) 움직임 추정 방법 및 장치

요약

움직임 추정 방법 및 장치에 관한 것으로서, 특히 주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률에 따라 후보벡터로 사용될 블록의 우선 순위를 결정하고, 우선 순위가 높은 블록의 움직임 벡터 순으로 후보벡터를 결정하며, 이때 동일한 값을 갖는 주위 벡터가 후보 벡터로 사용되지 않도록 함으로써, 현재 블록의 움직임과 가장 유사한 블록들의 움직임 벡터들을 현재 블록에 대한 후보벡터로 선택하고, 후보 벡터들이 항상 서로 다른 값을 가지므로 움직임 벡터의 정확도와 수렴 속도가 증가한다. 또한, 프레임마다 주위 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률을 계산하여 후보 벡터로 사용될 블록의 우선 순위를 갱신함으로써, 화면전환과 같이 이전 프레임들의 움직임 벡터의 분포와 다른 분포를 갖게 되는 경우에도 잘 대응할 수 있다.

대표도

도4

색인어

움직임 추정, 후보 벡터

명세서

도면의 간단한 설명

도 1은 종래의 3차원 회기형 움직임 추정의 개념도

도 2의 (a).(b)는 본 발명에 따른 주위 블록들의 움직임 벡터가 현재 블록과 일치하는 확률 측정의 실시 예를 보인 도면

도 3은 본 발명의 주위 블록들에 대한 우선 순위를 나타낸 예를 보인 도면

도 4는 본 발명에 따른 움직임 추정 방법을 나타낸 흐름도

도 5는 본 발명에 따른 움직임 추정 장치를 나타낸 구성 블록도

도 6은 도 5의 우선 순위 정보 산출부의 일 예를 보인 상세 블록도

도 7은 도 6의 적중률 갱신부의 상세 블록도

도 8은 본 발명에서 제안한 방식과 Hann 방식에 의해 구해진 움직임 벡터를 이용하여 이동 보상한 경우 PSNR를 나타낸 그래프

도면의 주요부분에 대한 부호의 설명

501 : 움직임 벡터 저장부	502 : 움직임 벡터 출력부
503 : 우선 순위 정보 산출부	504 : 후보 벡터 선택부
505 : 움직임 벡터 결정부	

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 비디오 디코더에 관한 것으로서, 특히 두 개의 영상 프레임사이에서 발생하는 움직임을 추정하는 방법 및 장치에 관한 것이다.

최근 가정용 TV 수신기가 고화질, 고품질화 되면서 수신기에는 두 개의 영상 프레임 사이에서 발생하는 움직임을 추정하고, 이에 의해 얻어진 움직임 벡터 정보를 이용하여 움직임 보상형 디-인터레이싱(de-interlacing), 움직임 보상형 프레임 율(frame rate) 변환, 그리고 움직임 보상형 잡음 제거와 같은 화질 향상 처리를 수행한다. 이와 같이 움직임 보상에 의한 처리를 위해서는 움직임 벡터의 추출을 위한 움직임 추정이 필수적으로 요구된다.

일반적으로 MPEG(Moving Picture Experts Group)등과 같은 영상 부호화에 사용되는 움직임 추정 방식은 현재 영상을 크기가 $N \times M$ 인 블록들로 나누고, 블록 정합(block matching) 방식에 의해서 현재 블록과 가장 유사한 블록의 위치를 이전 영상으로부터 구하는 블록 정합 방식(BMA: Block Matching Algorithm)을 사용한다. 특히 BMA 방식 중 가장 많이 사용되는 방식은 모든 가능한 후보 벡터(candidate motion vector)들에 대하여 정합을 수행하는 전역 탐색 블록 정합방식(FS-BMA: Full Search Block Matching Algorithm)이다. 이 방식은 움직임 보상된 신호에 의한 예측오차를 최소화시키므로 영상 부호화에 적용 시 수신단으로 전송해야 할 오차신호의 비트량을 크게 낮출 수 있다.

그러나, 상기 FS-BMA 방식은 움직임 벡터를 구하는데 많은 계산량을 요구한다. 예를 들어 움직임 벡터를 구하기 위한 탐색영역(Search Range)이 수평, 수직방향으로 $(\pm 10, \pm 10)$ 인 경우 $400(=20 \times 20)$ 번의 블록정합이 요구된다. 뿐만 아니라 TV 수신기에서 화질 향상을 위해 적용하는 움직임 보상형 디-인터레이싱, 움직임 보상형 프레임 율 변환 등의 처리에 사용되는 움직임 벡터는 예측오차를 최소화하기보다는 움직임 보상된 결과가 시각적으로 자연스러운 영상이 되도록 하는 것이 중요하다.

이러한 관점에서 움직임 추정을 위한 계산량을 크게 감소시키면서 주위 움직임 벡터들 사이의 상관성을 높이는 블록정합에 의한 3차원 회기형 움직임 추정(3-D recursive motion estimation) 방식이 Gerard de Haan 등에 의해서 제시되었다(Gerard de Haan, Paul W.A.C Biezen, Sub-pixel motion estimation with 3-D recursive search block-matching, Signal Processing: Image Communication(6)3 (1994)pp.229-239).

상기 3차원 회기형 움직임 추정 방식은 도 1에 도시한 바와 같이 현재 블록 X의 움직임 벡터 추정에 사용되는 후보벡터로 이미 움직임 추정이 수행된 블록 Sa, Sb, 그리고 T에 할당된 3개의 움직임 벡터를 사용한다. 여기서, Sa, Sb는 공간적 예측을 위한 현재 필드의 블록, T는 공간적-시간적 예측을 위한 이전 필드의 블록이다.

이때, 블록 X가 홀수번째 블록이면 Sa의 움직임 벡터에 업데이트(update) 움직임 벡터 \vec{v}_n 를 더한 값을 4번째 후보 벡터로 사용하고, 블록 X가 짝수번째 블록이면 Sb의 움직임 벡터에 업데이트 움직임 벡터

\vec{v}_n 를 더한 값을 4번째 후보 벡터로 사용한다. 그리고, 상기 4개의 후보 벡터들 중 예측오차가 가장 적

은 벡터를 블록 X의 움직임 벡터로 결정한다. 여기서, 업데이트 움직임 벡터 \vec{v}_n 은 실험적으로 미리 구해진 α 개의 움직임 벡터들로 구성된 집합으로 인덱스 n은 블록 X가 K번째 블록이면 $n = K\% \alpha$ 이다(%는 modular 연산).

따라서, 블록 X에 대해서 사용되는 업데이트 움직임 벡터 \vec{v}_n 은 α 개의 업데이트 움직임 벡터들 중 n번째 업데이트 움직임 벡터이다. 결국 이 방식은 탐색영역(Search Range)에 무관하게 항상 4개의 후보벡터만을 사용하므로 블록 당 4번의 블록 정합만을 요구한다. 또한 이 방식은 주위 블록들의 움직임 벡터를 후보벡터들로 사용하므로 주의 벡터들 사이의 상관성이 매우 높다.

발명이 이루고자하는 기술적 과제

하지만, 상기된 Gerard de Haan 등에 의해서 제시된 방식(이하 Hann 방식)의 경우, 신뢰할만한 움직임 벡터를 얻기 위해서는 움직임 벡터의 수렴시간이 필요하다. 즉, 현재 블록의 움직임 벡터는 주위 움직임 벡터들을 평가항으로써 얻어지므로 주위 벡터가 신뢰할 수 없는 경우 정확하지 못한 결과가 얻어진다. 따라서, 움직임 추정이 처음 이루어지는 초기 상황이나 화면전환(scene change)이 발생하는 경우는 주의 움직임 벡터들이 신뢰할 수 있는 벡터를 갖기 위한 수렴시간이 필요하다.

이와 같이 종래의 3차원 회기형 움직임 추정 방식(Haan 방식)은 매우 적은 계산량과 주위 벡터들 사이의 높은 상관성을 갖는 장점이 있지만, 적합하지 못한 후보 벡터 선정 방식에 의해서 충분히 빠른 수렴 속도와 정확한 움직임 벡터를 구하지 못하는 문제점이 있다.

따라서, 실제 TV 수상기에서 움직임 벡터 정보를 이용하여 움직임 보상형 디-인터레이싱, 움직임 보상형 프레임 율 변환, 그리고 움직임 보상형 잡음 제거와 같은 화질 향상 처리를 수행할 경우 보다 빠르고 정확한 움직임 벡터를 얻는 3차원 회기형 움직임 추정 방식의 개발이 요구된다.

본 발명은 상기와 같은 문제점을 해결하기 위한 것으로서, 본 발명의 목적은 보다 빠른 수렴시간과 정확한 움직임 벡터를 얻도록 하는 움직임 추정 방법 및 장치를 제공함에 있다.

발명의 구성 및 작용

상기와 같은 목적을 달성하기 위한 본 발명에 따른 움직임 추정 장치는, 주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률에 따라 후보 벡터로 사용될 블록의 우선 순위를 결정하는 단계

와, 우선 순위가 높은 블록의 움직임 벡터 순으로 K(K는 자연수)개의 후보 벡터를 결정하는 단계와, 상기 K개의 후보벡터들 중 하나의 후보 벡터에 업데이트 움직임 벡터를 더하고 그 값을 K+1번째 후보 벡터로 결정하는 단계와, 상기 K+1개의 후보벡터들 중 평가항수를 최소화시키는 벡터를 현재 블록의 움직임 벡터로 결정하는 단계를 포함하여 이루어지는 것을 특징으로 한다.

상기 우선 순위 결정 단계는 프레임마다 주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률을 계산하여 후보 벡터로 사용될 블록의 우선 순위를 갱신하는 것을 특징으로 한다.

상기 K개의 후보 벡터 결정 단계에서는 우선 순위대로 읽어들이는 움직임 벡터의 값이 현재까지 결정되어 있는 후보 벡터들의 값 중 어느 하나와 일치하면 상기 움직임 벡터를 후보 벡터에서 제외시키는 것을 특징으로 한다.

본 발명에 따른 움직임 추정 장치는, 움직임 벡터 저장부에 이전 프레임의 각 블록에 대한 움직임 벡터와 현재 프레임의 블록 중 이미 움직임 벡터가 산출된 블록의 움직임 벡터를 저장하고, 움직임 벡터 출력부는 상기 움직임 벡터 저장부에서 출력되는 움직임 벡터를 입력받고 우선 순위 정보에 따라 우선 순위가 높은 블록의 움직임 벡터부터 출력하며, 후보 벡터 선택부는 상기 움직임 벡터 출력부에서 우선 순위가 높은 순서대로 출력되는 움직임 벡터들을 입력으로 받아 K+1개의 서로 다른 값을 갖는 후보벡터들을 선택하는 것을 특징으로 한다. 이때, 우선 순위 정보 산출부는 주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률에 따라 후보 벡터로 사용될 블록의 우선 순위를 결정하고 그 정보를 상기 움직임 벡터 출력부로 제공하고, 움직임 벡터 결정부는 상기 후보 벡터 선택부에서 선택된 K+1개의 후보 벡터들 중 평가항수를 최소화시키는 벡터를 현재 블록의 움직임 벡터로 결정하는 것을 특징으로 한다.

상기 후보 벡터 선택부는 우선 순위가 높은 블록의 움직임 벡터 순으로 K(K는 자연수)개의 후보 벡터를 결정하고, 상기 K개의 후보벡터들 중 하나의 후보 벡터에 업데이트 움직임 벡터를 더하고 그 값을 K+1번째 후보 벡터로 결정하는 것을 특징으로 한다.

상기 우선 순위 정보 산출부는 매 프레임 주기마다 주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 적중률을 입력받아 주위의 블록이 현재 블록과 동일한 움직임 벡터를 가질 확률을 각각 계산하는 다수개의 적중률 갱신부와, 상기 각 적중률 갱신부의 출력을 비교하여 그 값이 큰 블록에 높은 우선 순위를 부여하는 비교기로 구성되는 것을 특징으로 한다.

본 발명은 주위 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률에 따라 후보벡터로 사용될 블록의 우선 순위를 결정하고, 우선 순위가 높은 블록의 움직임 벡터 순으로 후보벡터를 결정하며, 후보 벡터들이 항상 서로 다른 값을 가지도록 함으로써, 움직임 벡터의 정확도와 수렴 속도를 증가시키는데 있다.

또한, 프레임마다 주위 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률을 계산하여 후보 벡터로 사용될 블록의 우선 순위를 갱신하는데 있다.

본 발명의 다른 목적, 특징 및 잇점들은 첨부한 도면을 참조한 실시예들의 상세한 설명을 통해 명백해질 것이다.

이하, 본 발명의 바람직한 실시예를 첨부도면을 참조하여 상세히 설명한다.

통상, 현재 블록에 대한 움직임 벡터를 이미 움직임 벡터가 추정된 주위 블록의 벡터를 후보 벡터로 이용하여 구하는 경우 적합한 후보 벡터의 선택은 움직임 벡터의 정확도와 수렴 속도를 증가시킨다. 따라서, 만일 현재 블록의 움직임과 가장 유사한 블록들의 움직임 벡터를 현재 블록에 대한 후보벡터로 선택할 수 있다면 현재 블록에 대한 움직임 벡터의 정확도와 수렴 속도는 증가하게 된다.

도 2의 (a),(b)는 'table tennis'와 'mobile and calendar' 영상 시퀀스에 대하여 FS-BMA 방식으로 구한 움직임 벡터들로부터 현재 블록과 이미 움직임 추정이 수행된 주위 블록들의 움직임 벡터가 일치하는 확률을 나타낸 것으로, 현재 블록과 인접한 블록들이 동일한 움직임 벡터를 가질 확률이 높음을 보여준다(굵은 선으로 표시된 블록들).

따라서, 본 발명은 상기된 결과들로부터 현재 블록의 움직임 벡터와 일치하는 움직임 벡터를 가질 확률이 높은 블록의 우선 순위를 결정할 수 있다.

도 3은 주위 블록들에 대한 우선 순위를 나타낸 예를 보여 준다. 일 예로, 작은 수를 갖는 블록이 우선 순위가 높다. 이때, 우선 순위의 블록은 항상 고정시킬 수도 있고, 또는 도 6과 같이 적응적으로 결정할 수도 있다.

이와 같이 결정된 우선 순위에 따라 본 발명에서는 K개의 후보벡터를 결정하는데, 이때 동일한 후보벡터는 발생하지 않도록 함으로써 움직임 벡터의 정확도를 높인다. 이 과정을 예를 들어 설명하면 다음과 같다.

만일, 주위 블록의 움직임 벡터를 미리 우선 순위 순으로 나열한 결과가 아래의 수학적 식 1과 같고,

$$\begin{aligned} & \vec{v}_1 = (-5, -1), \quad \vec{v}_2 = (-5, 4), \quad \vec{v}_3 = (-4, 4), \quad \vec{v}_4 = (-5, 4), \\ & \vec{v}_5 = (-4, 4), \quad \vec{v}_6 = (-3, 3), \quad \vec{v}_7 = (-5, 4), \dots \end{aligned}$$

현재 블록의 움직임 벡터 추정에서 후보벡터로 사용되는 주위 블록의 움직임 벡터 수가 3개(즉, K=3)이

면, 최종적으로 후보 벡터로 선택되는 성분은 $\vec{v}_1 = (-5, 4)$, $\vec{v}_3 = (-4, 4)$, $\vec{v}_6 = (-3, 3)$ 이다. 왜냐하면 \vec{v}_2 와 \vec{v}_4 는 우선 순위가 높은 \vec{v}_1 과 동일한 값을 가지므로 후보 벡터에서 제외되고, \vec{v}_5 는 우선 순위가 높은 \vec{v}_3 와 동일한 값을 가지므로 후보 벡터에서 제외된다. 결국 K=3인 경우 선택된 후보 벡터의 집합 C는 $C = \{\vec{v}_1, \vec{v}_3, \vec{v}_6\} = \{\vec{v}_1, \vec{v}_3, \vec{v}_6\}$ 이다.

이와 같이 주위 블록의 움직임 벡터들 중 우선순위를 고려해서 구한 K개의 후보 벡터와 이들 중 하나의 후보 벡터에 업데이트 움직임 벡터 $\vec{v}^{(k)}$ 을 더한 움직임 벡터를 K+1번째 후보 벡터로 사용한다. 그리고, 현재 블록에 최종적으로 할당되는 움직임 벡터는 앞서 구한 K+1개의 후보 벡터 중 평가 함수(cost function)를 최소화하는 후보 벡터가 선택된다. 만일, 평가 함수가 하기의 수학식 2와 같은 움직임 보상된 블록과 현재 블록과의 차 신호(displaced block difference)의 절대값의 합을 최소화하는 후보 벡터가 현재 블록의 움직임 벡터 $\vec{v}^{(k)}$ 로 선택된다.

$$\vec{v}^{(k)} = \arg \min_{\vec{v} \in C} \sum_{i=1}^N |SAD(\vec{v}_i)|$$

여기서, 연산자 $SAD(\cdot)$ 는 현재 블록과 움직임 보상된 블록과의 차 신호의 절대값의 합을 출력하는 연산자이고, 집합 C는 선택된 K+1개의 후보 벡터의 집합이다.

도 4는 본 발명에서 제안한 3차원 회기형 움직임 벡터 추정 방법을 보여주는 흐름도이다.

일 예로, 움직임 추정에 사용될 수 있는 주위 블록의 수가 L개이고, 실제로 움직임 추정에 사용되는 후보 벡터의 수가 K+1 인 경우 처리과정은 다음과 같다.

우선, 본 발명에서 사용될 초기값 즉, $n=1$, $k=1$ 로 설정하고(단계 401), 후보 벡터로 사용될 수 있는 주위 블록의 움직임 벡터를 미리 결정된 우선 순위가 높은 순으로 읽는다(단계 402). 예를 들어 도 3의 경우 입력 순서는 각 블록에 기입된 숫자 순서와 동일하다. 즉, 작은 수를 갖는 블록이 우선 순위가 높다.

이때, 현재 상태가 k번째 후보 벡터를 결정하는 경우라면, 단계 403에서는 읽어 들인 움직임 벡터 $\vec{v}^{(k)}$ 이 현재까지 결정되어 있는 후보 벡터들 $C = \{\vec{v}^{(1)}, \dots, \vec{v}^{(k-1)}\}$ 과 일치하는 값인지를 확인한다. 만일 이전까지 결정된 k-1개의 후보 벡터 가운데 $\vec{v}^{(k)}$ 과 일치하는 벡터가 없다면 $\vec{v}^{(k)}$ 을 k번째 후보 벡터인 $\vec{v}^{(k)}$ 로 결정하고, k값을 하나 증가시킨다(단계 405). 한편, 일치하는 벡터가 있다면 n값을 하나 증가시킨 후(단계 404), 단계 402로 되돌아가 새로운 움직임 벡터를 읽어 들인다.

이와 같은 처리에 의해서 주위 블록의 움직임 벡터들 중 K개의 움직임 벡터를 후보 벡터로 결정한다($C = \{\vec{v}^{(1)}, \dots, \vec{v}^{(K)}\}$). 즉, 상기 단계 405가 수행되고 나서는 k가 K보다 큰지를 비교하여(단계 406), 크지 않다고 판별되면 아직 주위 블록의 움직임 벡터들 중 K개의 움직임 벡터를 후보 벡터로 결정한 경우가 아니므로 단계 404로 되돌아가 상기된 과정을 반복하고, 크다고 판별되면 주위 블록의 움직임 벡터들 중 K개의 움직임 벡터를 후보 벡터로 결정한 경우이므로 단계 407로 진행한다.

상기 407 단계에서는 K개의 후보 벡터들 중 하나의 벡터에 도 1에서 설명한 업데이트 움직임 벡터 $\vec{v}^{(k)}$ 을 더한 값을 K+1번째 후보 벡터로 결정한다($C = \{\vec{v}^{(1)}, \dots, \vec{v}^{(K)}, \vec{v}^{(K+1)}\}$). 여기서, $\vec{v}^{(K+1)} = \vec{v}^{(K)} + \vec{v}^{(k)}$ 이다.

상기 단계 407가 수행되면 마지막으로 상기와 같이 선택된 후보 벡터들 중 상기 수학식 2와 같은 평가 함수를 최소화시키는 벡터를 현재 블록의 움직임 벡터($\vec{v}^{(k)}$)로 결정한다(단계 408).

도 5는 본 발명에 따른 3차원 회기형 움직임 벡터 추정 장치의 구성 블록도로서, 이전 프레임의 각 블록에 대한 움직임 벡터와 현재 프레임의 블록 중 이미 움직임 벡터가 산출된 블록의 움직임 벡터를 저장하며 현재 블록의 위치 정보에 따라 현재 블록에 대해서 이웃한 위치에 있는 블록들의 움직임 벡터들을 출력하는 움직임 벡터 저장부(501), 우선 순위 정보에 따라 상기 움직임 벡터 저장부(501)에서 출력된 움직임 벡터들 중 우선 순위가 높은 블록의 움직임 벡터부터 출력하는 움직임 벡터 출력부(502), 움직임 벡터들의 우선 순위를 결정하여 상기 움직임 벡터 출력부(502)로 출력하는 우선 순위 정보 산출부(503), 상기 움직임 벡터 출력부(502)에서 우선 순위가 높은 순서로 출력되는 움직임 벡터들을 입력으로 받아 K+1개의 서로 다른 값을 갖는 후보 벡터들을 선택하는 후보 벡터 선택부(504), 및 상기 후보 벡터 선택부(504)에서 선택된 K+1개의 후보 벡터들 중 평가 함수를 최소화시키는 벡터를 현재 블록의 움직임 벡터로 결정하는 움직임 벡터 결정부(505)로 구성된다. 이때, 상기 움직임 벡터 결정부(505)에서 결정된

현재 블록의 움직임 벡터는 움직임 벡터 저장부(501)로 피드백되어 다음 블록의 움직임 벡터 결정에 사용되는 주위 움직임 벡터로 사용된다.

이와 같이 구성된 도 5에서 상기 움직임 벡터 저장부(501)는 이전 프레임의 각 블록에 대한 움직임 벡터와 현재 프레임의 블록 중 이미 움직임 벡터가 산출된 블록의 움직임 벡터를 저장한다. 그리고, 상기 움직임 벡터 저장부(501)는 현재 블록의 위치 정보에 따라 현재 블록에 대해서 이웃한 위치에 있는 블록들의 움직임 벡터들을 움직임 벡터 출력부(502)로 출력한다. 즉, 상기 움직임 벡터 저장부(501)에서 출력되는 움직임 벡터들은 우선 순위 정보 산출부(503)의 우선순위 정보에 따라 우선순위가 높은 블록의 움직임 벡터부터 움직임 벡터 출력부(502)를 통해 후보 벡터 선택부(504)로 출력된다. 여기서, 상기 우선순위 정보 산출부(503)는 항상 고정된 우선 순위를 출력할 수도 있고, 도 6에서 설명할 적응적인 방식으로 우선 순위를 결정하여 출력할 수도 있다.

상기 후보 벡터 선택부(504)는 상기과 같이 우선순위가 높은 순서로 나열된 움직임 벡터들을 움직임 벡터 출력부(502)를 통해 입력받아 도 4에서 설명한 바와 같이 K+1개의 서로 다른 값을 갖는 후보벡터들을 선택한 후($\{V_1, V_2, \dots, V_{K+1}\}$), 움직임 벡터 결정부(505)로 출력한다.

상기 움직임 벡터 결정부(505)에서는 K+1개의 후보 벡터들 중 평가함수를 최소화시키는 벡터를 현재 블록의 움직임 벡터(V')로 결정한다. 이때, 상기 움직임 벡터 결정부(505)에서 결정된 현재 블록의 움직임 벡터(V')는 다시 움직임 벡터 저장부(501)에 저장되어 다음 블록의 움직임 벡터 결정에 사용되는 주위 움직임 벡터로 사용된다.

도 6은 상기 우선 순위 정보 산출부(503)의 상세 블록도이고, 도 7은 도 6의 각 적응을 갱신부의 상세 블록도로서, 매 프레임 마다 적응적으로 움직임 벡터들의 우선순위를 결정하는 예를 보이고 있다.

도 6에서 각 적응을 갱신부는 한 프레임 기간동안 주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 적응률을 입력받는다. 예를 들어, 현재 블록 위치가 (x,y,m+1)인 경우 $I(x,y,m+1)$ 은 이전 프레임에서 현재 블록의 위치와 동일한 위치의 블록이 현재 블록과 움직임 벡터가 일치한 비율을 나타낸다. 따라서, 매 프레임 주기마다 새로운 적응률을 입력받고, 각 적응을 갱신부에서는 다음과 같은 수학적 3의 연산을 수행하여 주위의 블록이 현재 블록과 동일한 움직임 벡터를 가질 확률을 계산한다.

$$O(x,y,m) = \alpha \times I(x,y,m) + (1-\alpha) \times O(x,y,m-1)$$

여기서, α 의 범위는 $0 \leq \alpha \leq 1$ 인데 α 값이 클수록 프레임마다 변화하는 움직임 벡터 분포의 변화에 민감하게 반응한다. 즉, 도 7은 상기된 수학적 3을 하드웨어로 구현한 것이다.

그리고, 각 적응을 갱신부의 출력은 비교기에서 비교되어 값이 큰 블록에 높은 우선순위를 부여한다.

이와 같은 처리에 의하여 화면 전환등이 발생하여 현재 프레임에 움직임 벡터 분포가 이전 프레임들의 분포와 다른 경우에도 잘 적응할 수 있다.

도 8은 본 발명에서 제안한 방식과 Hann 방식에 의해 구해진 움직임 벡터를 이용하여 이동 보상의 경우 PSNR(Peak Signal to Noise Ratio)를 나타낸 것으로 본 발명의 방식이 정확도와 수렴 속도를 향상시킬 수 있다. 이 실험의 경우는 고정된 우선순위를 적용하였고 현재 블록의 움직임 벡터 산출에 사용된 후보 벡터의 수는 Hann의 경우와 마찬가지로 4개이다.

따라서, 본 발명은 실제 TV 수상기에서 움직임 벡터 정보를 이용하여 움직임 보상형 디-인터레이싱, 움직임 보상형 프레임 을 변환, 그리고 움직임 보상형 잡음 제거와 같은 화질 향상 처리를 수행할 경우 매우 효과적이다.

발명의 효과

이상에서와 같이 본 발명에 따른 움직임 추정 방법 및 장치에 의하면, 주위 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률에 따라 후보벡터로 사용될 블록의 우선 순위를 결정하고, 우선 순위가 높은 블록의 움직임 벡터 순으로 후보벡터를 결정하며, 이때 동일한 값을 갖는 주위 벡터가 후보 벡터로 사용되지 않도록 함으로써, 현재 블록의 움직임과 가장 유사한 블록들의 움직임 벡터들을 현재 블록에 대한 후보벡터로 선택하고, 후보 벡터들이 항상 서로 다른 값을 가지므로 움직임 벡터의 정확도와 수렴 속도가 증가하게 된다.

또한, 프레임마다 주위 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률을 계산하여 후보 벡터로 사용될 블록의 우선 순위를 갱신함으로써, 화면전환과 같이 이전 프레임들의 움직임 벡터의 분포와 다른 분포를 갖게 되는 경우에도 잘 대응할 수 있다. 특히, FS-BMA 등과 비교하면 계산량을 크게 감소시키므로 구현이 간단해지는 효과가 있다.

이상 설명한 내용을 통해 당업자라면 본 발명의 기술 사상을 일탈하지 아니하는 범위에서 다양한 변경 및 수정이 가능함을 알 수 있을 것이다.

따라서, 본 발명의 기술적 범위는 실시예에 기재된 내용으로 한정되는 것이 아니라 특허 청구의 범위에 의하여 정해져야 한다.

(57) 청구의 범위

청구항 1

주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률에 따라 후보 벡터로 사용될 블록의 우선 순위를 결정하는 단계;

우선 순위가 높은 블록의 움직임 벡터 순으로 K(K는 자연수)개의 후보 벡터를 결정하는 단계;

상기 K개의 후보벡터들 중 하나의 후보 벡터에 업데이트 움직임 벡터를 더하고 그 값을 K+1번째 후보 벡터로 결정하는 단계; 그리고

상기 K+1개의 후보벡터들 중 평가함수를 최소화시키는 벡터를 현재 블록의 움직임 벡터로 결정하는 단계를 포함하여 이루어지는 것을 특징으로 하는 움직임 추정 방법.

청구항 2

제 1 항에 있어서, 상기 단계에서 주위 블록의 움직임 벡터는

이전 프레임의 각 블록에 대한 움직임 벡터와 현재 프레임의 블록 중 이미 움직임 벡터가 산출된 블록의 움직임 벡터들인 것을 특징으로 하는 움직임 추정 방법.

청구항 3

제 1 항에 있어서, 상기 우선 순위 결정 단계는

프레임마다 주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 확률을 계산하여 후보 벡터로 사용될 블록의 우선 순위를 갱신하는 것을 특징으로 하는 움직임 추정 방법.

청구항 4

제 1 항에 있어서,

상기 단계에서 평가 함수가 다음의 식에 적용되면, 움직임 보상된 블록과 현재 블록과의 차 신호의 절대값의 합을 최소화하는 후보벡터가 현재 블록의 움직임 벡터(\vec{V}_i)로 결정되는 것을 특징으로 하는 움직임 추정 방법.

$$V_i = \arg \min_{V_i \in C} \sum_{j=1}^N |SAD(\vec{V}_i - \vec{V}_j)|$$

여기서, 연산자 $SAD(\cdot)$ 는 현재 블록과 움직임 보상된 블록과의 차신호의 절대값의 합을 출력하는 연산자이고, 집합 C는 선택된 K+1개의 후보 벡터의 집합임.

청구항 5

이전 프레임의 각 블록에 대한 움직임 벡터와 현재 프레임의 블록 중 이미 움직임 벡터가 산출된 블록의 움직임 벡터를 저장하며 현재 블록의 위치 정보에 따라 현재 블록에 대해서 이웃한 위치에 있는 블록들의 움직임 벡터들을 출력하는 움직임 벡터 저장부;

우선 순위 정보에 따라 상기 움직임 벡터 저장부에서 출력된 움직임 벡터들 중 우선 순위가 높은 블록의 움직임 벡터부터 출력하는 움직임 벡터 출력부;

움직임 벡터들의 우선 순위를 결정하여 상기 움직임 벡터 출력부에 우선 순위 정보를 출력하는 우선 순위 정보 산출부;

상기 움직임 벡터 출력부에서 우선 순위가 높은 순서대로 출력되는 움직임 벡터들을 입력으로 받아 K+1개의 서로 다른 값을 갖는 후보벡터들을 선택하는 후보 벡터 선택부; 그리고

상기 후보 벡터 선택부에서 선택된 K+1개의 후보 벡터들 중 평가함수를 최소화시키는 벡터를 현재 블록의 움직임 벡터로 결정하며, 상기 결정된 움직임 벡터를 상기 움직임 벡터 저장부에 피드백시켜 저장하는 움직임 벡터 결정부를 포함하여 구성되는 것을 특징으로 하는 움직임 추정 장치.

청구항 6

제 5 항에 있어서, 상기 후보 벡터 선택부는

우선 순위가 높은 블록의 움직임 벡터 순으로 K(K는 자연수)개의 후보 벡터를 결정하고, 상기 K개의 후보벡터들 중 하나의 후보 벡터에 업데이트 움직임 벡터를 더하고 그 값을 K+1번째 후보 벡터로 결정하는 것을 특징으로 하는 움직임 추정 장치.

청구항 7

제 6 항에 있어서,

상기 K개의 후보 벡터 결정시 우선 순위대로 읽어들이는 움직임 벡터의 값이 현재까지 결정되어 있는 후보 벡터들의 값 중 어느 하나와 일치하면 상기 읽어들이는 움직임 벡터를 후보 벡터에서 제외시키는 것을 특징으로 하는 움직임 추정 장치.

청구항 8

제 5 항에 있어서, 상기 우선 순위 정보 산출부는

매 프레임 주기마다 주위 블록의 움직임 벡터가 현재 블록의 움직임 벡터와 일치하는 적중률을 입력받아

주위의 블록이 현재 블록과 동일한 움직임 벡터를 가질 확률을 각각 계산하는 다수개의 적중률 갱신부와,

상기 각 적중률 갱신부의 출력을 비교하여 그 값이 큰 블록에 높은 우선 순위를 부여하는 비교기로 구성되는 것을 특징으로 하는 움직임 추정 장치.

청구항 9

제 8 항에 있어서,

상기 각 적중률 갱신부에서는 다음의 식을 적용하여 하드웨어를 구성하고, 주위의 블록이 현재 블록과 동일한 움직임 벡터를 가질 확률($O(x,y,m)$)을 계산하는 것을 특징으로 하는 움직임 추정 장치.

$$(\partial_t \chi - \chi - m) = 0, \quad (\partial_t \chi - \chi_1 m) = 0, \quad 1 = \chi + (\partial_t \chi - \chi_1 m) = 1.$$

여기서, α 의 범위는 $0 \leq \alpha \leq 10$ 이고, $I(x, y, m)$ 은 이전 프레임에서 현재 블록의 위치와 동일한 위치의 블록이 현재 블록과 움직인 벡터가 일치한 비율을 나타냄.

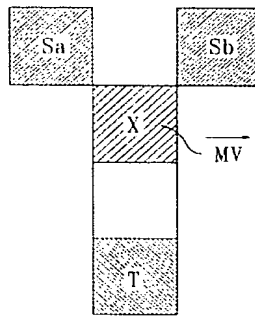
청구항 10

제 5 항에 있어서, 상기 우선 순위 정보 산출부는

현재 블록의 움직임 벡터와 일치할 확률이 높은 주위 블록의 움직임 벡터에 대해 우선 순위를 고정시키는 것을 특징으로 하는 움직임 추정 장치.

도전

521



X:current block

Sa,Sb:blocks in current field for spatial prediction

T:block in previous field for spatio-temporal prediction

update motion vector(MV)

$$U_1 = \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} 3 \\ 0 \end{pmatrix}, \begin{pmatrix} -3 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0.25 \end{pmatrix}, \begin{pmatrix} 0 \\ -0.25 \end{pmatrix}, \begin{pmatrix} 0.25 \\ 0 \end{pmatrix}, \begin{pmatrix} -0.25 \\ 0 \end{pmatrix} \right\}$$

도면2

이전 프레임					현재 프레임				
0.344	0.390	0.419	0.409	0.372	0.331	0.414	0.459	0.444	0.394
0.421	0.479	0.511	0.495	0.438	0.449	0.536	0.595	0.558	0.482
0.470	0.536	0.571	0.539	0.465	0.529	0.637			
0.450	0.497	0.518	0.487	0.430					
0.392	0.423	0.432	0.409	0.368					

현재 블록

(a)

이전 프레임					현재 프레임				
0.759	0.774	0.780	0.772	0.761	0.759	0.776	0.782	0.774	0.762
0.770	0.791	0.804	0.787	0.771	0.771	0.793	0.807	0.789	0.774
0.790	0.823	0.892	0.823	0.797	0.792	0.827			
0.769	0.790	0.805	0.788	0.773					
0.749	0.766	0.771	0.761	0.751					

현재 블록

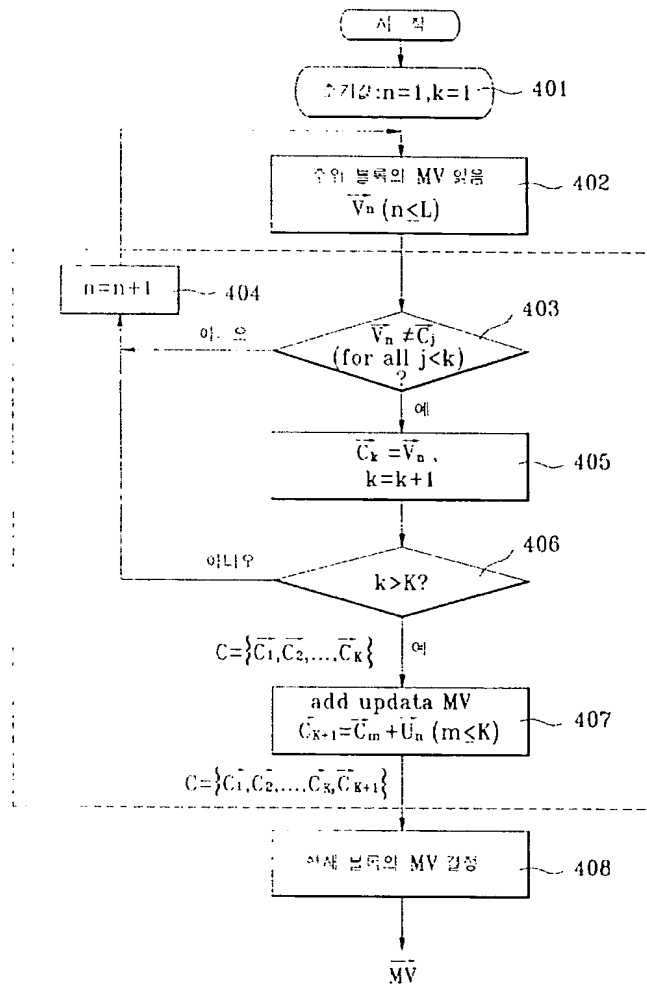
(b)

도면3

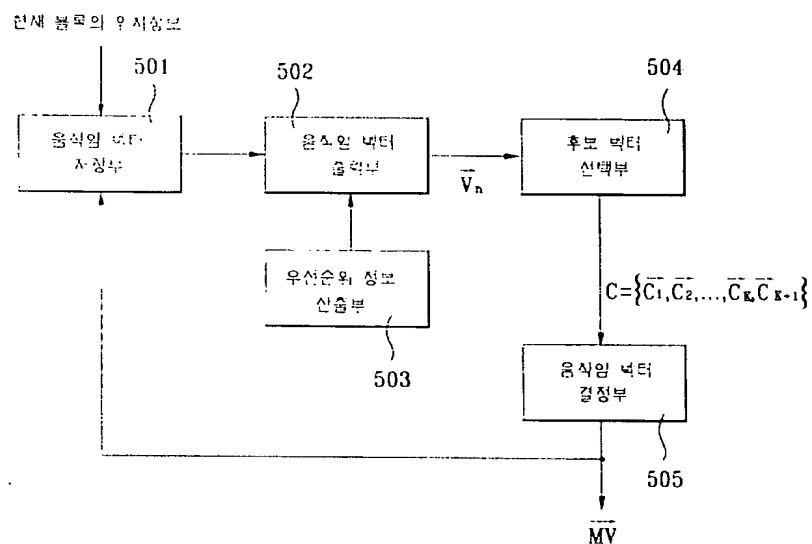
이전 프레임			현재 프레임		
12	7	3	8	5	9
4	1	3	2		
11	6	10			

현재 블록

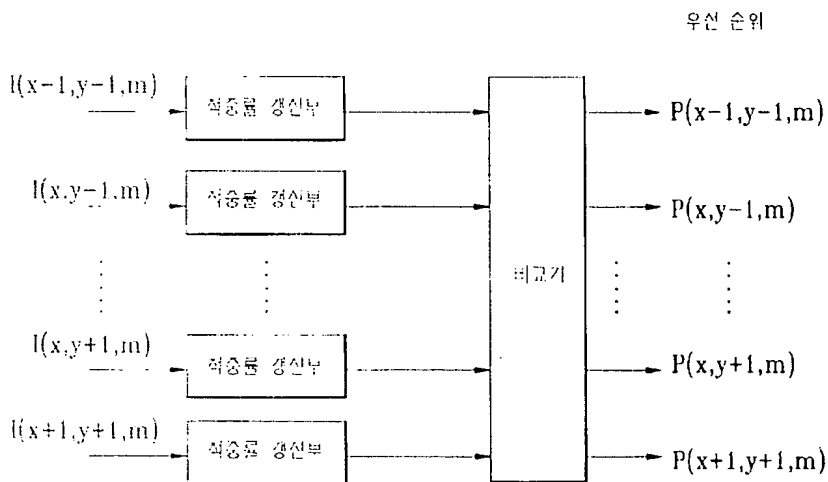
도면4



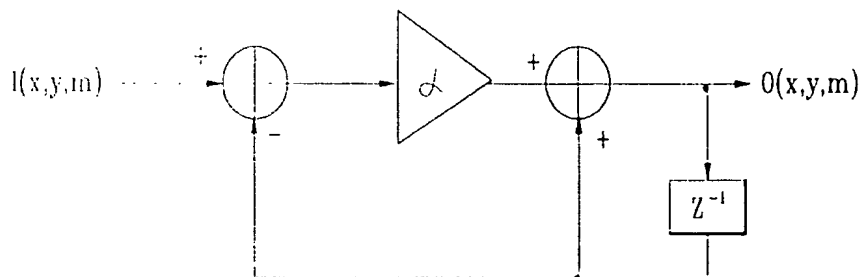
도 215



5216



597



도면8

